UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/764,961	01/26/2004	Beat Stamm	14984.34	9335

47973 7590 10/06/2006

WORKMAN NYDEGGER/MICROSOFT
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UT 84111

EXAMINER	
WOODS, ERIC V	
ART UNIT	**PAPER NUMBER**
2628	

DATE MAILED: 10/06/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	10/764,961	STAMM ET AL.
	Examiner	**Art Unit**
	Eric Woods	2628

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 02 February 2006.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-13 and 15-20 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☒ Claim(s) 11-13 and 15-19 is/are allowed.

6)☒ Claim(s) 1-7,9,10 and 20 is/are rejected.

7)☒ Claim(s) 8 is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on 02 February 2006 is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☒ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Response to Arguments

Applicant's arguments, see pages 1-3, filed 2 February 2006, with respect to the rejection(s) of various claims under various statutes have been fully considered and are persuasive.

Examiner thanks applicants for the prompt submission of the various Information Disclosure Statements. By doing so, applicant has fully complied with 37 CFR 1.56 with regards to the issues raised by examiner in the previous Office Action.

The objections to paragraphs [0022] and [0038] of the specification stand withdrawn in view of the applicant's amendment to the specification to correct the deficiencies in [0022] and the pointing out where the specification supports the element not previously noted in [0038].

The objection to the drawings stands withdrawn in view of applicant's amendments.

The objections to the claims stand withdrawn in view of applicant's amendments to the claims to correct those deficiencies and the addition of claim 5.

The rejections of claims 1 and 20 under 35 USC 112, second paragraph, as indefinite stand withdrawn in view of applicant's amendments to the claims to correct those deficiencies.

The rejections of claims 11-19 under 35 USC 103(a) stand withdrawn in view of applicant's amendments to the claims and cancellation of claims.

Claim 15 was canceled.

However, upon further consideration, a new ground(s) of rejection is made in view of various references against claims 1-10 and 20 as set forth below.

It is pointed out that the claim 20 as currently written is non-statutory (see attached Interview Summary). An after-final amendment to correct that deficiency **only** would be entered if sent.

Examiner notes that applicant argues that Stamm and Collins only provide translation between TrueType and Postscript (as part of an example below). However, it is irrelevant where the control points come from, or any other details. The key point is – does the combination of the references perform the stated functions, regardless of whether or not it performs other, additional ones? Examiner's position is that the combination applied below constitutes at least functionally equivalent steps in the overall process, which therefore meets the limitations below.

It is also noted that the Shamir reference could be applied to the instant claims as performing the first, added step rather effectively.

Allowable Subject Matter

Claims 11-13 and 15-19 would be allowable if they were rewritten to overcome the rejections under 35 USC 101 below. Specifically, the subject matter indicated allowable in dependent claim 14 was moved to independent claim 11. Claims 12-13 and 15-19 depend upon claim 11 and are thusly allowable. Claim 14 was canceled.

As such, independent claim 11 would be allowed, as would be dependent claims 12-13 and 15-19, if the above issues concerning 35 USC 101 were resolved. Reasons for allowance were provided in the last Office Action.

Claim 8 would be allowable if rewritten in independent form to include all of the limitations of the base claim and any intervening claims.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1, 11, and 20 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Specifically, the claims lack a practical, concrete, and tangible application. The claims do not result in either a physical transformation or a concrete, tangible application, since the result of the process is only more data. The resultant hinting language instructions are not displayed to a user, nor are the optimized results used to generate character data that is displayed. Therefore, the claims only manipulate data within a computer and represent the quintessential abstract idea.

Claims 2-10 and 12-19 are rejected as not correcting the deficiencies of their parent claim(s).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been

obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

Claims 1-2, 7, 9-10, and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Stamm (US 6,249,908 B1) in view of Collins et al (US 5,817,714 A)('Collins') and DeRose et al (US PGPub 2001/0002131 A1)('DeRose').

As to claims 1 and 20 (computer-implemented method and computer program product),

In a computing system that has access to a set of control points, the set of control points for generating an outline of a graphical object, the outline being utilized to determine how the graphical object is rendered, the position of some portions of the outline potentially being constrained to pre-determined locations, a method for using a font-hinting language to represent an iterative solution to a constraint, the method comprising: (Stamm teaches that a font hinting language inherently provides the preamble – TrueType and higher level abstractions are taught in 1:37-54. Secondly, the above cited paragraph recites the use of control

points, and of holding points at certain values and with certain relationships, and clearly it is specified the typographer sets forth the outline of the font.)(Collins also uses TrueType – 2:50-65, and in Collins the approximations of the outline are constrained by the existing font being converted from – e.g. 15:60-16:15, where the new, low level constraints are bounded by the location of the device-independent points)

-Identifying features of the graphical object represented by a set of control points expressly representing strokes to identify a more complex constraint that cannot be natively expressed based on the vocabulary of the font hinting language; (Stamm recognizes that glyphs are composed of contours – that is, a glyph is represented by a compound data structure, consisting of one or more contours, where these contours consist of one or more control points, where the relationships are expressed based on relationships between control points (3:45-4:30). Glyphs are inherently outlines (1:13-36), where these are composed of glyph segments, which are composed of control points and their relationships (see Figure 4b). Therefore, identifying glyph segments – **by whatever means this occurs** – still constitutes identifying features, where each segment represents a stroke. A stroke is (3:1-7) a component of the glyph, e.g. a glyph contour / segment (9:59-10:7, see Figure 4d, 10:19-63). Therefore, identifying contours / segments of the glyph constitutes strokes, where the relationships between the control points represent strokes as defined therein – where the terms "expressly representing" are construed as open-ended, in that as long as strokes are identified by the segments, it does not exclude other

implementations.)(Collins – in 2:50-65, the use of multiple languages such as Adobe Postscript (Type 1) and Intellifont description languages are taught. Further, it is known that in PostScript page description language that it is a high-level language that describes page layout, fonts, and non-character shapes – 23:40-50. Clearly, constraints in such a language are more advanced – e.g. are "complex" as defined by the applicant, and a cubic Bezier curve (such as that used in Postscript Type 1 fonts) is more complex than the quadratic Bezier curve allowed in the TrueType language.) Therefore, any conversions by Collins from PostScript to TrueType take higher-order Bezier curves (e.g. cubic ones), since they are not native to TrueType); clearly, such high level constructs as third-order cubic Bezier curves cannot be expressed in TrueType)

-Accessing a more complex constraint that cannot be natively expressed based on the vocabulary of the font-hinting language, the more complex constraint constraining at least a portion of the outline; (Stamm teaches that the Type Man Talk high-level abstraction of the TrueType language is cited in the above paragraph. Further, in 3:11-35 it is stated the Stamm invention generates high-level hinting information for a font, where such information is translated or interpreted to and from graphical format so that the user can see such information. Clearly, it is intended that the user generate complex programs (e.g. features not natively supported at a low-level of TrueType or similar program languages)(see the Abstract and 18:45-65. In that passage (18:45-65), it is stated that experienced font programmers can alter the font using a high-level programming language but still see the graphical representations of their work.

Further, the "high-level" language mentioned clearly constitutes a high-level abstraction of the low level TrueType instructions or similar as cited above.)(Collins teaches accessing a predefined font description, which uses a plurality of outline characters to define segments of the shape of a character in a font description language – see 47:35-39. Collins then teaches that a new font description of outline segments is generated from the predefined one in a new font description language. In 2:50-65, the use of multiple languages such as Adobe Postscript (Type 1) and Intellifont description languages are taught. Further, it is known that in PostScript page description language that it is a high-level language that describes page layout, fonts, and non-character shapes – 23:40-50. Clearly, constraints in such a language are more advanced – e.g. are "complex" as defined by the applicant, and a cubic Bezier curve (such as that used in Postscript Type 1 fonts) is more complex than the quadratic Bezier curve allowed in the TrueType language.)

-Decomposing the more complex constraint into a plurality of simpler constraints that can be natively expressed based on the vocabulary of the font-hinting language; and (In Stamm, clearly, the high-level instructions or complex constraints are translated into low-level instructions (e.g. native TrueType commands). See 7:55-8:4, where it is expressly taught that within Figure 2, the graphical representation 200 is translated or converted by process 215 into high-level language hinting instructions 225 which are then compiled to low-level TrueType instructions (unlabeled step in Figure 2, but the box labeled 'TrueType Instructions' is present). The compilation process from high-level language into

low-level (or native) commands or constraints clearly constitutes the recited decomposition, and it is well established in the art that TrueType is a font-hinting language. As stated in the opening portion of the specification, e.g. 1:37-54, the hints can be specified in either the low level language or the higher level abstraction, so clearly the simpler constraints would be those of the commands in the low level language as recited in the clause above.)(Collins clearly teaches translating one font language to another, and when the first font language is Adobe Postscript Type 1 or 3 with cubic Bezier splines and the target language is TrueType, which only supports quadratic Bezier curves, it would be obvious that a more complex function would be translated to two less complex ones, e.g. it is well known in the art that in order to replicate an 'S' character, one cubic Bezier curve would be sufficient, since it has two inflection points to change direction, whereas implementing it with quadratic Bezier curves would require two curves to generate the same result. Clearly, transferring between one and other would clearly involve decomposing higher level instructions into lower-level ones in this particular case)

-Representing each of the simpler constraints in corresponding font-hinting language instructions that are iteratively processed to at least approximate a solution to the more complex constraint. (Stamm clearly teaches that the more complex language is decomposed to more simple instructions.)(Collins clearly teaches that the translated instructions are approximate – see 47:57-64, where the shape of the outline is approximated between device-independent points with new segments bounded by such points. Such approximations are part of

modeling a new font to mimic the translated one (47:48-58). In 3:45-55, the idea of stretching an existing font to approximate another one is mentioned but disparaged. The solution present by Collins involves (15:60-16:14) choosing tangent points that are not labeled as a corner and are between a curve and a line segment, where the circumstances are such that it is unlikely that a single cubic Bezier curve could approximate the shape of both. New segments are formed based on various techniques taught therein.)(DeRose clearly teaches in the abstract that the system is directed to more efficiently modeling objects in computer graphics. In [0026] it is stated that the system first produces a two-dimensional mesh. Note that it is irrelevant that this two-dimensional mesh is applied to a three-dimensional object insofar as the positions of the control points on the mesh are concerned (see [0027-0028]). Now, there are equations that describe the positions of the control points on the mesh, as established in [0031-0033]. Regardless of the three-dimensional import of those equations, the fact remains that they are applied to a **two-dimensional** mesh. This mesh has a two-dimensional texture applied to it [0038]. Now, in [0044-0045], it is taught that the energy function is **restricted to two dimensions** [0045], where it can be implemented on surface defined by NURBS (non-uniform rational B-splines, which can include Bezier patches). The point of this exercise is that in [0046-0047] it is clear that the minimization of the energy function is done in **two dimensions**. This minimization is done using Newton's method or other standard numerical technique for minimization (see [0047]), where in [0051-0052] it is taught that as is well known in the art, Newton's method is iterative and is

used to converge the equations to a more refined approximation, where in [0051] “a few iterations generally suffice.” Clearly, this method is used **approximate** a best fit ([0028-0029, 0039-0040, 0045-47, and 0051] to the energy function, which very clearly meets the standards of the claim. Clearly, the method works to **position the control points on the mesh in such a way as to minimize the energy function** (see [0051]).)

Clearly, the equations referred to in DeRose [0051] that are used to define the energy function could just as easily be used to approximate a best fit to the original font and could be used to position the control points cited in Collins above to minimize the fit between the segments and points of the outline character in the original font description or hinting language in Collins and the translated version in the second font or character description language. Collins states that the method also applies to cases where high-level languages contain items other than fonts (23:40-50), as in page description languages, and in Stamm et al, a paper is listed in the “Other Publications” sections called “Glyphs: flyweight objects for user interfaces”, where that only serves to emphasize the point that glyphs are graphical objects. Collins teaches the key linkage, where the approximated segments are created (15:60-16:14) in a manner that is designed to approximate the curves and features of the font that they came from. The equations described therein could constitute the energy functions of DeRose that would clearly then be minimized using a standard numerical technique, such as Newton’s method. DeRose is merely used as an example of a well-known technique in the art, that approximate solutions can be greatly improved by

iterating through equation systems defining the approximations until they converge to a satisfactory degree, and examiner asserts that using iterative techniques are well known in the art, and further takes Official Notice to that effect, since it is well known that the advantage of the numerical methods like Newton is that (as stated in DeRose), they generally only require a few iterations and produce satisfactory results. Further, DeRose clearly teaches in 18:10-17 that lines or Bezier curves are used to perform the approximation under certain circumstances, and that line or curve fitting techniques are used to find the approximation. It is well known in the art that Newton's method can be used to perform curve and line fitting to get a good approximation.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the systems of Stamm and Collins, as Collins clearly allows a system to create a version of a font that is similar to one that is desired to be displayed, but that the target system does not possess (as in on the Internet – see Collins Abstract) and the portable format will allow the target system to merely download sets of shapes and create the new font at the operating system level (4:47-5:30), where the motivation is to allow new text formats that are more interesting to the user. The system of Stamm, while allowing a font designer to create new fonts, does not expressly allow the creation of new fonts based on portable or cross-platform formats, and does not explicitly discuss translating between font hinting languages. The system of Collins allows for the translation of fonts generated in one font language (e.g.

Postscript) into another (e.g. TrueType), which would be another advantage that Stamm would gain by the incorporation of Collins.

Motivation for combination of Stamm and Collins with DeRose is provided above, with the combination of DeRose serving to speed up the approximations used by Collins to generate new font elements and the like.

Further, Stamm and Collins are clearly analogous art with the instant application, and Collins and DeRose are also directed to the same problem-solving area.

As to claim 2, the energy functions of DeRose as cited in claim 1 can be exponential and power functions as stated n for example [0045-0047] and the like. Furthermore, the curve and line fitting methods of Collins clearly apply, where it is well known that a curve fitting function is nonlinear, and nonlinear functions clearly involve power or exponentials in some way. Finally, a cubic Bezier curve in Postscript being transformed to a quadratic Bezier curve in TrueType clearly involves some kind exponential and/or power function mapping. Motivation for combination is taken from the rejection to claim 1 as above.

As to claim 7, the system of Stamm clearly teaches in 9:59-6:18 that each control point (in the data structure) has a freedom direction data field to indicate the direction that the point can be moved, and further has both move and delta exception fields, which allow the control point to be moved a certain distance regardless of ppem variations or vice versa. This clearly teaches that control points can be moved. Further, Stamm sets forth a graphical environment where the user can manipulate the characteristics of a font, which will then be translated

into high-level language commands. Basically, since the combination of Stamm and Collins was explained in the rejection to claim 1, the decomposition of high-level functional language into TrueType low-level assembly commands would necessarily result in the decomposition of complex operations into low-level constraints that only can involve moving one point (e.g. imposing a simple fixed or relative distance constraint). Those are the only kind that the low-level TrueType format allows, and as set forth above, the system of Stamm clearly has provisions for moving control points one at a time. As such, the rejection to claim 1 is incorporated by reference.

As to claim 9, the system of Stamm clearly utilizes low level TrueType instructions as set forth in the rejection to claim 1, which is incorporated by reference.

As to claim 10, Collins clearly teaches in 18:10-27 that the system calculates the number of times a curve has to be subdivided into two before the worst error is less than one half ORU. As stated above in the rejection to claim 1, which is incorporated by reference, it would be obvious to use an iterative method, since that method is always bounded by a finite constraint and would be an improvement over the recursive methods of Collins that could lead an infinite loop and high memory consumption. Both techniques are well known in the computer science art (iteration and recursion) and can be used interchangeably. In any case, Newton's method (as cited in DeRose) is iterative and is incorporated by reference. The concept of having a loop iterate a number of times until it reaches a certain specified tolerance is comparable to the concept

of having a function recurse until it converges to a certain tolerance band around a solution (and DeRose teaches that usually a few iterations of Newton's method is sufficient in any case, where 'sufficient' clearly entails the idea of a tolerance band.

As stated in the rejection to claim 1, Collins approximates a solution using curve- or line-fitting techniques (18:10-27), where iterating instead of recursing has already been covered, and the system of Stamm converts high level instructions to low level ones, and clearly Collins teaches in the claims (the 47:x – y reference locations in the rejection to claim 1) that such translations from one language to another end up altering the outline in one way or another, and in the figures that seems to be reflected. Obviously, the final output object will have a tolerance within the band mentioned by DeRose or as set forth above by Collins. Clearly, the final result of the process would be a pixilated representation of the graphical object (e.g. a bitmap) as taught by Collins in 2:64-3:20 and 10:27-45, where bitmap representations are put into the portable font format, and it is known that the final output of the font to a screen or a printer will require it to be converted to a raster graphics (e.g. pixilated) representation. Motivation is incorporated form the rejection to claim 1. Also, such output fonts are output to monitor 47 of Stamm Figure 1, which inherently requires a pixilated representation.

Claims 3 and 4 are rejected as unpatentable over Stamm in view of Collins and DeRose as applied to claim 1 above, and further in view of Rappoport et al (WO 98/36630)('Rappoport').

As to claim 3,

The method as recited in claim 1, wherein accessing a more complex constraint that cannot be natively expressed based on the vocabulary of the font-hinting language comprises accessing a constraint that requires a plurality of control points to be moved simultaneously.

First of all, the system of Stamm clearly teaches in 9:59-6:18 that each control point (in the data structure) has a freedom direction data field to indicate the direction that the point can be moved, and further has both move and delta exception fields, which allow the control point to be moved a certain distance regardless of ppem variations or vice versa. This clearly teaches that control points can be moved. Further, Stamm sets forth a graphical environment where the user can manipulate the characteristics of a font, which will then be translated into high-level language commands. The idea of being able to move multiple control points, while not being explicit, is strongly suggested. The other two references do not per se teach this limitation, although DeRose teaches moving control points on the mesh in response to minimization of the energy function, which would seem to comprise moving multiple control points simultaneously.

Rappoport clearly teaches how one external parameter (see abstract) can be linked to multiple constraints (see Figure 1, where the parameters 24 are linked to constraints 22, and Rappoport clearly teaches changing external font

parameters in the Abstract and in the locations stated above. Further, in 15:1-10 and in other locations, a user interface to simultaneously adjust several external font parameters is taught, and these clearly effect how the font is rendered, for example, one is listed in Figs. 8A-8D, and for example the width of a stem is changed in Figs. 4A-4B, as noted in the first paragraph, and it moves the associated control points on the stem to the left and the right, which obviously involves moving multiple control points. Basically, examiner's position is that when changing any of the major dimensions of a font with global parameters, multiple control points must *prima facie* be moved simultaneously to effect the required scaling.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Rappoport with Stamm, Collins, and DeRose, for several reasons, one being that applicant supplied a relevant article by Rappoport concerning the same subject matter, which is an implicit admittance of relevancy; secondly, Rappoport allows the user to manipulate many constraints on a glyph or a font generally in a graphical manner (see 15:1-10) in a way that would give the hypothetical font designer or typographer utilizing the system of Stamm the ability to further manipulate all the glyphs in the advanced hinting system taught therein in a manner that would allow simultaneous control over several relevant parameters of the font, thusly shortening the design cycle (see for example 15:1-10 and other relevant locations).

As to claim 4,

The rejection to claim 3 is incorporated by reference. Further, the Stamm reference does inherently teach this limitation for the reason that it teaches the use of letters in a font that are Roman, e.g. the letter 'z' as given as an example in the instant application, which would obviously have circularly dependent constraints in the very nature of its configuration. Further, Rappoport also teaches in Figures 11-14 the use of the letter 'A' which has diagonal components that would clearly also have circular constraints as defined in the specification, where a circular constraint would merely consist of control points with a fixed distance between them, where each effected the other, which could very easily be the case, where in Rappoport Figure 14, multiple constraints involving distances in the letter 'A' are shown, where clearly some of those would be connected in such a way that circular dependencies would result. Clearly, since the existence of such dependencies has been proven, it would be obvious that when the font is accessed, circularly dependent constraints are prima facie accessed.

Claim 5 is rejected under 35 USC 103(a) as unpatentable over Stamm in view of Collins and DeRose as applied to claim 1 above, and further in view of Shamir et al ("Extraction of Typographic Elements from Outline Representation of Fonts" – from applicant's IDS).

Stamm, etc., do not expressly teach identifying features of the graphical object expressly representing serifs to identify a more complex constraint that cannot be natively expressed based on the vocabulary of the font hinting

language. Shamir (page C-267, upper right corner) teaches the use of a high-level stem ladder having plural levels, **where Adobe Type 1 fonts only have three stems**. Therefore, it would have been obvious that the same principle could be applied to serifs, where a plurality of them could exist. Also, Shamir expressly teaches using control points to recognize serifs (page C-265, section 4.4). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the system of Stamm, Collins, and DeRose to accommodate higher-order glyph constructs consisting of a plurality of stems, serifs, or the like, which clearly are not natively supported by the lower-level languages, which is extremely beneficial to Oriental fonts (CJK)(Shamir entirety of page C-267).

Claim 6 is rejected under 35 U.S.C. 103(a) as unpatentable over Stamm in view of Collins and DeRose as applied to claim 1 above, and further in view of Weisstein (Weisstein, Eric W. "Taylor Series". From *MathWorld*—A Wolfram Web Resource, 1999).

Claim 6 is similar to claim 1 except that the decomposition of a power series into discrete elements is recited. It is well known in the art that a Taylor series can be used to approximate a function. The first page of the Weisstein reference provides a Taylor series approximation of an exponential function by providing the expansion for it, where the expansion is most assuredly a plurality of terms of a power series. It would have been obvious to use a Taylor series to do so because it allows a quick approximation with low error for a power series.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric Woods whose telephone number is 571-272-7775. The examiner can normally be reached on M-F 7:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ulka Chauhan can be reached on 571-272-7782. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Eric Woods

August 16, 2006

Ulka Chauhan

ULKA CHAUHAN
SUPERVISORY PATENT EXAMINER